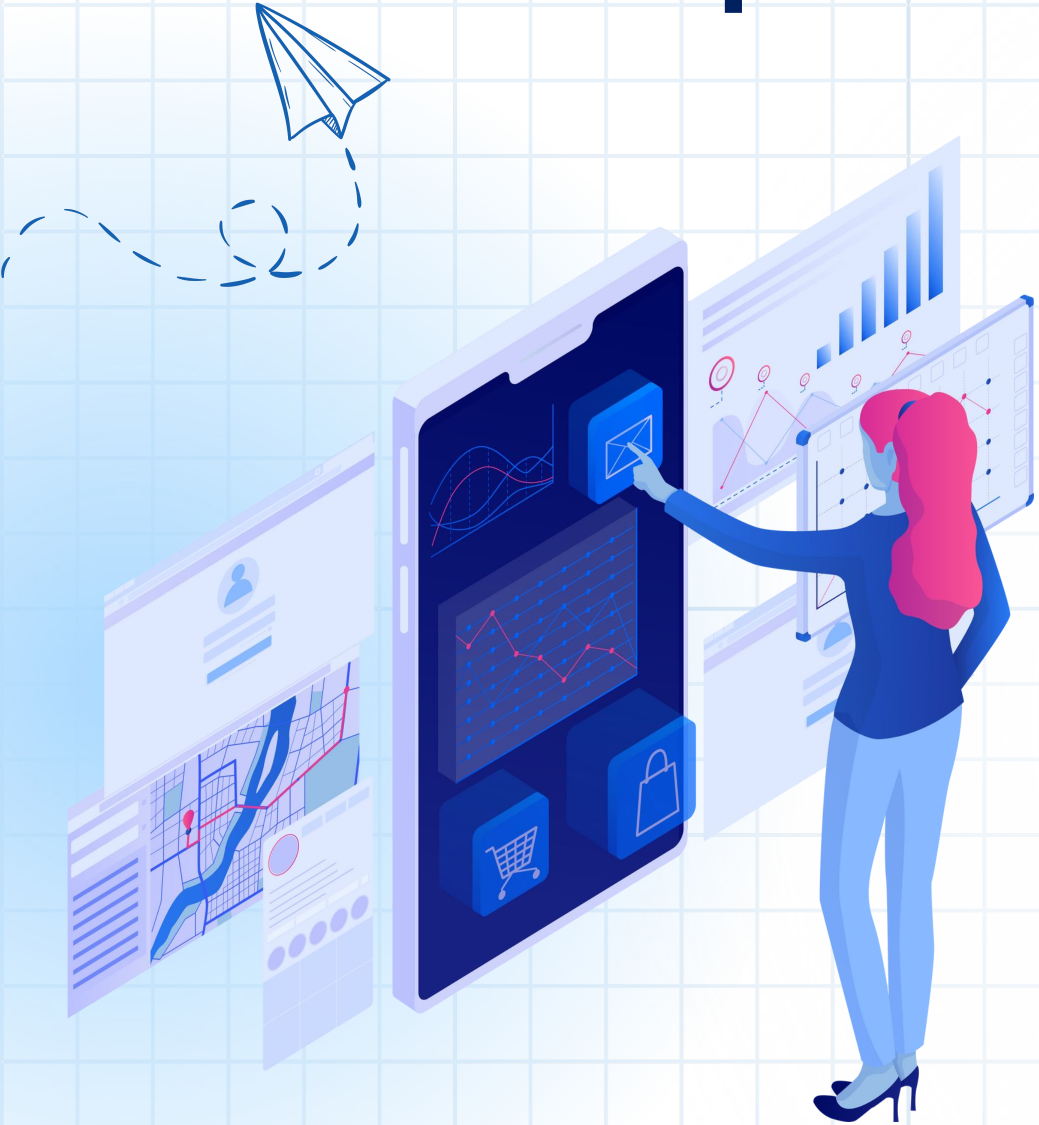
BS./BSc.

in Applied AI and Data Science

Linear algebra and numerical analysis

Module 3.4: Recap Vector space and linear independence

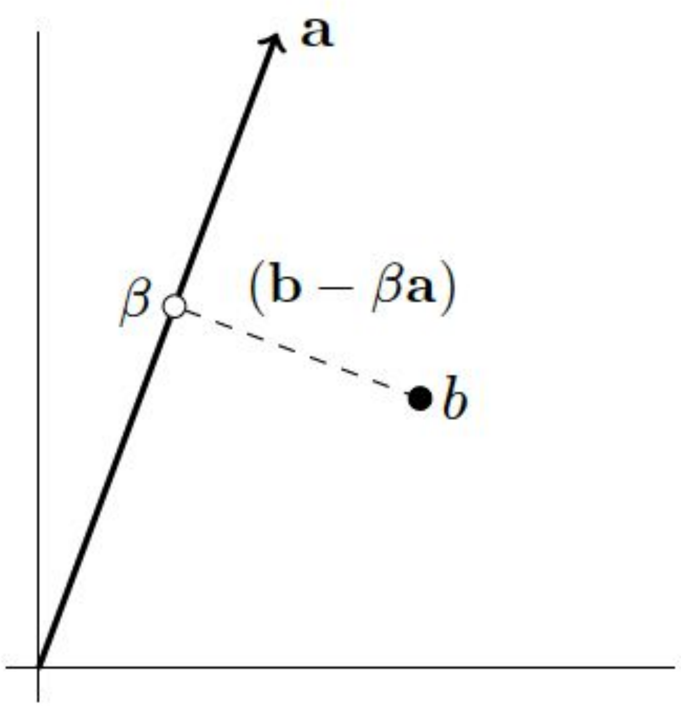


1. Review of Vector space and linear independence
2. Geometric interpretation of Projection in vector space
3. Orthogonal matrices
4. Gram-Schmidt procedure for orthogonalization

Module 3.5: Projections in R^2

We are going to discover a formula for projecting a point onto a line. From there, we can generalize the formula to projecting one space onto another space. Don't worry, it's easier than you might think.

We start with a vector a, a point b not on a, and a scalar such that   a is as close to b as possible without leaving a. Figure below depicts the situation. (Because we are working with standard position vectors, equating coordinate points with vectors is possible.)



Geometric interpretation of projection

There is a geometric interpretation of the previous example helps in solving algebraic equations

Let's recap: We have established that the closest projection of b onto a involves the line that meets **a** at a right angle, and we have an expression for that line.

Now our goal is to figure out the value of beta. The geometric approach already provided the key insight; we just need to translate that into algebra. In particular, vectors **(b− a)** and a are orthogonal, meaning they are perpendicular:

$$(\mathrm{b} - \beta \mathbf{a}) \perp \mathbf{a}$$

And that in turns means that the dot product between them is zero. Thus, we can rewrite

$$(\mathrm{b} - \beta \mathbf{a})^{\mathrm{T}} \mathbf{a} = 0$$

Orthogonal Projections of a point into line

We have discovered a beautiful formula for projecting a point onto a line indispensable for Eigen decomposition and widely used data science.

Orthogonal projection of a point onto a line

$$\mathbf{a}^{\mathrm{T}}(\mathbf{b} - \beta\mathbf{a}) = 0$$

$$\mathbf{a}^{\mathrm{T}}\mathbf{b} - \beta\mathbf{a}^{\mathrm{T}}\mathbf{a} = 0$$

$$\beta\mathbf{a}^{\mathrm{T}}\mathbf{a} = \mathbf{a}^{\mathrm{T}}\mathbf{b}$$

$$\beta = \frac{\mathbf{a}^{\mathrm{T}}\mathbf{b}}{\mathbf{a}^{\mathrm{T}}\mathbf{a}}$$

$$\mathrm{proj}_a(b) = \frac{\mathbf{a}^{\mathrm{T}}\mathbf{b}}{\mathbf{a}^{\mathrm{T}}\mathbf{a}}\mathbf{a}$$

*Note that dividing both sides of the equation by **$a^T a$** is valid because it is a scalar quantity. The technical term for this procedure is projection: We are projecting **b** onto the subspace defined by vector **a**.*

Powered by Futurense

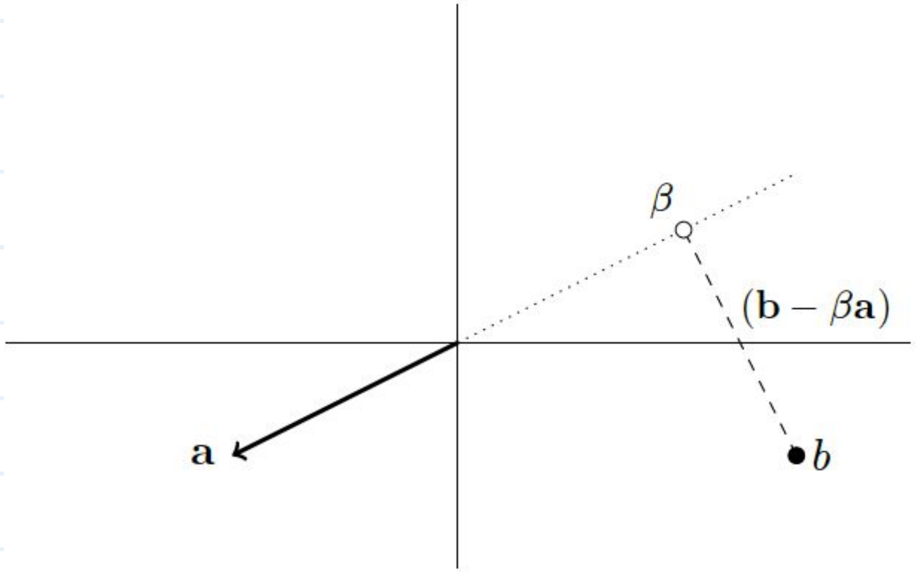
Question 2

Given the following vector and point. Could the point be projected onto a line

$$\mathbf{a} = \begin{bmatrix} -2 \\ -1 \end{bmatrix}, \quad b = (3, -1)$$

Solution

$$proj_a(b) = \frac{\begin{bmatrix}-2\\-1\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}3\\-1\end{bmatrix}}{\begin{bmatrix}-2\\-1\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}-2\\-1\end{bmatrix}}\begin{bmatrix}-2\\-1\end{bmatrix} = \frac{-6+1}{4+1}\begin{bmatrix}-2\\-1\end{bmatrix} = -1\begin{bmatrix}-2\\-1\end{bmatrix}$$



The point is "behind" the line, which will project negatively onto the vector. Notice that beta = –1. Thus, we are projecting "backward" onto the vector. This makes sense when we think of a as being a basis vector for a 1D subspace that is embedded in R^2.

Orthogonal matrices

I will begin by introducing you to orthogonal matrices. An orthogonal matrix is a special matrix important for several decompositions, including QR, eigen decomposition, and singular value decomposition.

The letter Q is often used to indicate orthogonal matrices.
Orthogonal matrices *have two properties:*
Orthogonal columns
All columns are pair-wise orthogonal.
Unit-norm columns
The norm (geometric length) of each column is exactly 1.

Module 3.6: Mathematical definition of orthogona

We can translate those two properties into a mathematical expression (remember that $\langle \mathbf{a}, \mathbf{b} \rangle$ is an alternative notation for the dot product):

$$\langle \mathbf{q}_i, \mathbf{q}_j \rangle = \begin{cases} 0, & \text{if } i \neq j \\ 1, & \text{if } i = j \end{cases}$$

What does that mean? It means that the dot product of a column with itself is 1 while the dot product of a column with any other column is 0. That's a lot of dot products with only two possible outcomes. We can organize all of the dot products amongst all pairs of columns by premultiplying the matrix by its transpose. Remember that matrix multiplication is defined as dot products between all rows of the left matrix with all columns of the right matrix; therefore, the rows of $\mathbf{Q}^{\mathrm{T}}$ are the columns of $\mathbf{Q}$.

The matrix equation expressing the two key properties of an orthogonal matrix is simply marvelous:

$$\mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{I}$$

The expression $\mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{I}$ is amazing. Really, it's a big deal.

Orthogonal matrices some examples

Why is it a big deal?

Because Q^T *is a matrix that multiplies* Q *to produce the identity matrix, that's the exact same definition as the matrix inverse.*

$$\mathbf{Q}^T\mathbf{Q} = \mathbf{Q}\mathbf{Q}^T = \mathbf{Q}^{-1}\mathbf{Q} = \mathbf{Q}\mathbf{Q}^{-1} = \mathbf{I}$$

Thus, the inverse of an orthogonal matrix is its transpose. That's crazy cool because the matrix inverse is tedious and prone to numerical inaccuracies, whereas the matrix transpose is fast and accurate.

Do such matrices really exist in the wild, or are they mere figments of a data scientist's imagination?

$$\mathbf{Q} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \qquad \mathbf{Q} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

Pythoning to confirm orthogonality

Please take a moment to confirm that each column has unit length and is orthogonal to other columns. Then we can confirm in Python:

```
Q1 = np.array([ [1,-1],[1,1] ]) / np.sqrt(2)
Q2 = np.array([ [1,2,2],[2,1,-2],[-2,2,-1] ]) / 3

print( Q1.T @ Q1 )
print( Q2.T @ Q2 )
```

Question 3

Consider the following matrix. Take a moment to prove to yourself that it is an orthogonal matrix.

$$\mathbf{Q} = \frac{1}{3}\begin{bmatrix} 2 & 1 \\ -2 & 2 \\ 1 & 2 \end{bmatrix}$$

Solution

Pre- and post-multiplying this matrix by its transpose shows that Q^T is the left inverse of Q.

$$\mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \frac{1}{9}\begin{bmatrix} 2 & -2 & 1 \\ 1 & 2 & 2 \end{bmatrix}\begin{bmatrix} 2 & 1 \\ -2 & 2 \\ 1 & 2 \end{bmatrix} = \frac{1}{9}\begin{bmatrix} 9 & 0 \\ 0 & 9 \end{bmatrix}$$

$$\mathbf{Q}\mathbf{Q}^{\mathrm{T}} = \frac{1}{9}\begin{bmatrix} 2 & 1 \\ -2 & 2 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} 2 & -2 & 1 \\ 1 & 2 & 2 \end{bmatrix} = \frac{1}{9}\begin{bmatrix} 5 & -2 & 4 \\ -2 & 8 & 2 \\ 4 & 2 & 5 \end{bmatrix}$$

Recap

Orthogonalization of vectors is a fundamental process in linear algebra that aims to create a set of orthogonal vectors from a given set of linearly independent vectors. Here's a summary of the key concepts:

Orthogonality:

- ❖ Two vectors are orthogonal if their dot product is zero, meaning they are perpendicular to each other.
- ❖ A set of vectors is orthogonal if every pair of vectors within that set is orthogonal.

Orthogonalization Goal: The process transforms a set of linearly independent vectors into a set of orthogonal vectors that span the same subspace.

Orthonormalization: If, in addition to being orthogonal, each vector in the set has a length (or norm) of 1, the set is called orthonormal.

Coming up next...............

- Concepts of *QR decomposition*
- The Gram-Schmidt procedure is a way to transform a nonorthogonal matrix into an orthogonal matrix.

Thank you